

Comparative Study of Pressure-Correction and Godunov-Type Schemes on Unsteady Compressible Cases.

Frédéric Archambeau*, Jean-Marc Hérard, Jérôme Laviéville

Department of Fluid Dynamics, Power Generation and Environment, EDF-R&D, France

11th December 2008

Abstract

Two pressure-correction algorithms are studied and compared to an approximate Godunov scheme on unsteady compressible cases. The first pressure-correction algorithm sequentially solves the equations for momentum, mass and enthalpy, with sub-iterations which ensure conservativity. The algorithm also conserves the total enthalpy along a streamline, in a steady flow. The second pressure-correction algorithm sequentially solves the equations for mass, momentum and energy without sub-iteration. This scheme is conservative and ensures the discrete positivity of the density. Total enthalpy is conserved along a streamline, in a steady flow. It is numerically verified that both pressure-correction algorithms converge towards the exact solution of Riemann problems, including shock waves, rarefaction waves and contact discontinuities. To achieve this, conservativity is compulsory. The two pressure-correction algorithms and the approximate Godunov scheme are finally compared on cases with heat source terms: all schemes converge towards the same solution as the mesh is refined.

Keywords: compressible flow; pressure-correction algorithm; Godunov scheme; finite volume conservative scheme; positivity; total enthalpy conservation

Nomenclature

Roman upper case

E	total energy, $E = \varepsilon + \frac{1}{2} u^2$	$m^2 s^{-2}$
H	total enthalpy, $H = h + \frac{1}{2} u^2$	$m^2 s^{-2}$
L	1D domain length	m
L_Φ	domain length where an energy source term is applied	m
P	pressure	$kg m^{-1} s^{-2}$
S	area of the interfaces of the mesh	m^2
T	temperature	K

*Corresponding author. Address: Department of Fluid Dynamics, Power Generation and Environment, EDF-R&D, 6 Quai Watier, 78400 Chatou, France. E-mail address: frederic.archambeau@edf.fr

Roman lower case

c	speed of sound $c = \sqrt{\partial_\rho P _s}$	$m s^{-1}$
c_P	$c_P = \partial_T h _P$	$m^2 s^{-2} K^{-1}$
c_V	$c_V = \partial_T \varepsilon _\rho$	$m^2 s^{-2} K^{-1}$
eps	relative precision for the iterative solver	-
$ncel$	number of cells of the mesh	-
ntg	number of momentum-enthalpy-pressure global iterations	-
$nthm$	number of enthalpy-pressure sub-iterations	-
h	enthalpy $h = H - \frac{1}{2} u^2$	$m^2 s^{-2}$
Q	mass flux (ρu)	$kg m^{-2} s^{-1}$
q	mass flux (ρu)	$kg m^{-2} s^{-1}$
q_{ac}	mass flux for advective terms	$kg m^{-2} s^{-1}$
q_f	mass flux at cell face	$kg m^{-2} s^{-1}$
r	constant for perfect diatomic gas	$m^2 s^{-2} K^{-1}$
s	entropy	$m^2 s^{-2} K^{-1}$
t	time	s
u	velocity	$m s^{-1}$
x	space coordinate	m

Greek upper case

Δx	distance between the centres of two adjacent cells	m
Φ	magnitude of the energy source term	$kg m^{-1} s^{-3}$
Ω	volume of a cell, here $\Omega = \Omega_i$	m^3
Ω_i	volume of cell i , here $\Omega_i = S \Delta x$	m^3
Ω_{tot}	volume of the computational domain	m^3

Greek lower case

γ	$\gamma = \frac{c_P}{c_V}$	-
ε	internal energy $\varepsilon = E - \frac{1}{2} u^2$	$m^2 s^{-2}$
ρ	density	$kg m^{-3}$

Superscripts

n	time step number	-
	-	-

Subscripts

i	mesh cell number	-
$i + 1/2$	mesh interface number	-
k	enthalpy-pressure sub-iteration number	-
m	momentum-enthalpy-pressure global iteration number	-

1 Introduction

Being able to calculate compressible flows is important for aeronautic applications, but also for power generation industry. Indeed, compressible flows may be encountered in quite a wide variety of situations. Steam flow in turbines is a quite usual example. It is also crucial to be able to deal

with compressible flows for the safety studies in the nuclear power-plants (for instance: steam-water flows in the primary circuit or complex gas mixtures in the containment). The present paper has been motivated by a safety study on a large electrical power transformer involving a compressible flow with a very large energy source term independent of density [6], [7]. Indeed, under specific conditions, an electric arc may strike, accompanied by a very strong and local Joule effect heat source term (typically 10^{10} W m^{-3}) and by a modification of the fluid properties at high temperature (typically $30,000 \text{ K}$): the flow becomes locally compressible and the pressure rise resulting from such conditions may deteriorate the structure of the power transformer.

We want to demonstrate that so-called pressure-correction algorithms enable us to retrieve meaningful solutions of compressible Navier-Stokes equations, even when shock solutions occur. As a first step, the present considerations are limited to Euler equations.

Actually, in order to solve Euler equations with shockwaves, Godunov-type schemes are usually considered and have already proved to provide very good approximate solutions on any mesh size ([12], [23]). There is a huge amount of literature devoted to the latter framework, which of course is not recalled here.

To address flows at all speeds, extensions of compressible flow solvers to low- or zero-Mach-number flows have been proposed. For example, in [21], the authors propose an extension of compressible flow solvers to variable-density low-Mach-number flows. In [25], an advective upwind splitting method (AUSM) is used with a pressure velocity coupling through an artificial diffusion added to the mass flux (only steady-state equations are considered in this contribution). In [8], methods for extending AUSM based schemes to operate at all speeds are developed (only steady-state flows are investigated in this contribution).

On the other hand, few papers have been published which address the suitability of pressure-correction algorithms for Euler equations with shockwaves. In [16], the authors present a coupled pressure- and temperature-correction algorithm constructed so that Mach-uniform accuracy and efficiency are obtained: various steady-state test cases are considered. In [2], Mach-uniform accuracy and efficiency are also hunted for, using a pressure-correction method on a staggered grid: the method is applied to steady cases and to two unsteady configurations (a shock tube and a nozzle flow). An improvement of this method has been provided in [24], with respect to the conservativity in space of the energy equation (the conservativity in time has not been specifically assessed). However, for pressure-correction methods applied to transient flows with shockwaves, there seems to be no published work including a quantitative error analysis of the convergence towards the exact solution as the mesh is refined.

Of course, we do not expect pressure-correction algorithms to be more efficient than Godunov-type schemes for shock solutions. Our concern is to show that the domain of application of some existing pressure-correction schemes covers compressible flows with shocks: even if such schemes are not optimal, they are stable and converge towards the exact solution.

Hence, we present two pressure-correction algorithms and compare them on the basis of several unsteady test-cases for which an exact solution is available (double rarefaction wave, contact discontinuity, double shock wave, shock tubes and a test-case including a heat source term). The evolution of the L^1 -norm of the error obtained with these schemes is displayed as the mesh is refined (see [9] for a comparison with other finite volume schemes). To the authors' knowledge, this is an

original contribution of the present paper.

In particular, the present paper examines the impact of conservativity in time (and in space, of course): it is demonstrated numerically that if this property is not perfectly insured, pressure-correction schemes may remain stable but do not necessarily converge towards the right shock solution. This may remain undetected in complex industrial studies.

Eventually, both pressure-correction algorithms and an approximate Godunov scheme are used to calculate an unsteady test-case including a strong local and constant heat source term, for which no exact solution exists. The effect of the mesh refinement is studied.

The three algorithms considered in the present study have been implemented in one dimension of space, with the sole purpose of solving the specific set of equations considered here. Their main characteristics are summarized hereafter (see table 1 for a summary).

- The first pressure-correction algorithm, called NLK, sequentially solves the equations of momentum, mass and enthalpy, with sub-iterations to ensure conservativity. The algorithm also conserves the total enthalpy along a streamline, in a steady flow when no dissipation/source term is present.
- The second pressure-correction algorithm, called SLK, sequentially solves the equations of mass, momentum and energy without sub-iteration. By construction, this scheme is conservative. Provided a condition on the time step value is satisfied, the scheme ensures the discrete positivity of the density and of scalar variables when no source term is present. The algorithm also preserves the total enthalpy along a streamline, in a steady flow when no dissipation/source term is present.
- The third and last algorithm, VFRoe-ncv, is an approximate conservative Godunov scheme [4].

	NLK	SLK	VFRoe-ncv
Conservativity	yes	yes	yes
Preservation of the total enthalpy	yes	yes	yes
Positivity of the density and of the scalar variables		yes	

Table 1: *Summary of the properties of the algorithms (NLK is assumed to be operated with enough sub-iterations).*

The outline of the present paper is the following. The set of continuous equations is first detailed and the discrete forms of the two pressure-correction algorithms are indicated (a brief description of the VFRoe-ncv algorithm is provided in the appendix, referring to [4]). Then, the test-cases are described and convergence results are presented. In particular, the behaviour of the algorithms on the test-case with a constant heat source term is discussed.

2 Governing equations and grid related notations

We will consider the following 1D-problem, for $t_{min} \leq t \leq t_{max}$ and $x_{min} \leq x \leq x_{max}$:

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) & = 0 \\ \partial_t(\rho u) + \partial_x(\rho u u) + \partial_x P & = 0 \\ \partial_t(\rho H) + \partial_x(\rho u H) & = \partial_t P + \Phi \end{cases} \quad (1)$$

This system stands for the standard Euler equations including a heat source term Φ independant of time.

The equation of state corresponds to perfect gas thermodynamics:

$$P = \rho r T \quad (2)$$

The specific enthalpy, the internal energy, the isobaric and isochoric specific heats and the square of the sound speed are recalled below:

$$\begin{cases} h = \frac{\gamma P}{(\gamma - 1)\rho} \text{ and } \varepsilon = \frac{P}{(\gamma - 1)\rho} \\ c_P = \frac{r\gamma}{\gamma - 1} \text{ and } c_V = \frac{r}{\gamma - 1} = c_P - r \\ c^2 = \frac{\gamma P}{\rho} \text{ and } \beta = \rho^\gamma \end{cases} \quad (3)$$

Except otherwise stated, the following constants are used:

$$\gamma = 1.4 \text{ and } r = \frac{R}{M} = \frac{8.31434}{28.8 \cdot 10^{-3}} \quad (4)$$

The system of continuous equations (1) is used as such for the discretization of the NLK algorithm. For the SLK algorithm, the system is written using the total energy instead of the total enthalpy:

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) & = 0 \\ \partial_t(\rho u) + \partial_x(\rho u u) + \partial_x P & = 0 \\ \partial_t(\rho E) + \partial_x(\rho u E + u P) & = \Phi \end{cases} \quad (5)$$

Before providing the details pertaining to the discretization, some notations are introduced, in association with the numbering of the cells and of the interfaces of the finite volume mesh.

A colocated finite volume approach is used for all schemes, on a uniform 1D mesh containing an even number of cells, $ncel$. The cells are numbered from $i = 1$ to $i = ncel$. All cells have the same volume, denoted $\Omega_i = \Omega$. The distance between the centres of two adjacent cells is $\Delta x = \frac{L}{ncel}$, with L the length of the computational domain.

The interface between two adjacent cells i and $i + 1$ is referred to as “face $i + 1/2$ ”. The two boundary faces are referred to as faces “ $1/2$ ” and “ $ncel + 1/2$ ” respectively. All the faces j , from $j = 1/2$ to $j = ncel + 1/2$, have the same area, $S = 1$.

3 NLK algorithm

3.1 Algorithm overview

The scheme is first order in time. At each time step, a global iterative procedure couples the velocity step and the enthalpy-pressure step; this iterative procedure stops when the variations of the divergence of the mass flux are sufficiently small (or when the maximum number of iterations has been reached). At each of these global iterations, the solution proceeds as follows:

- the velocity is predicted, with an explicit pressure gradient and a semi-implicit convection term; the mass flux is updated using the so-called “Rhie and Chow interpolation” defined in section 3.3.1;
- a sub-iterative procedure couples enthalpy and pressure corrections; at each sub-iteration of this procedure:
 - the total enthalpy is calculated and the density is updated;
 - the enthalpy-pressure sub-iterative procedure stops (and the next global iteration/time step starts):
 - * if the maximum number of enthalpy-pressure sub-iterations has been reached
 - * or if at least one sub-iteration has been completed and the discrete mass equation is satisfied (given a predetermined user-defined precision requirement);
 - the pressure correction is determined and is used to update the mass flux and the velocity, the pressure and the density.

3.2 Discretization in time

3.2.1 Global coupling

For $m = 1$ to $m = ntg$ (unless otherwise stated, $ntg = 1$), one iterates over the next five steps (from “Velocity prediction” to “End of the subcycles”), starting with:

$$Q_{m=1}^{n+1} = q^n \quad (6)$$

For $n = 0$, u^0 , P^0 and H^0 are given, and we take $q^0 = \rho^0 u^0$ with ρ^0 from the equation of state.

3.2.2 Velocity prediction

The predicted velocity u_m^* is calculated from:

$$\rho^n \frac{u_m^* - u^n}{\Delta t} - u_m^* \partial_x Q_m^{n+1} + \partial_x (q^n u_m^*) = -\partial_x P^n \quad (7)$$

3.2.3 Enthalpy-pressure coupling

For $k = 1$ to $k = nthm$ (unless otherwise stated, $nthm = 2$), one iterates over the next two steps (enthalpy and pressure correction), starting with:

$$(u, H, P, q)_{m,k=1}^{n+1} = (u_m^*, H^n, P^n, \rho^n u_m^*) \quad (8)$$

3.2.4 Enthalpy

The total enthalpy $H_{m,k+1}^{n+1}$ is calculated from:

$$\rho^n \frac{H_{m,k+1}^{n+1} - H^n}{\Delta t} - H_{m,k+1}^{n+1} \partial_x q_{m,k}^{n+1} + \partial_x (q_{m,k}^{n+1} H_{m,k+1}^{n+1}) = \frac{P_{m,k}^{n+1} - P^n}{\Delta t} + \Phi \quad (9)$$

Then, an updated density $\rho_{m,k+1/2}^{n+1}$ is calculated:

$$\rho_{m,k+1/2}^{n+1} = \frac{\gamma P_{m,k}^{n+1}}{(\gamma - 1)(H_{m,k+1}^{n+1} - \frac{1}{2} u_{m,k}^{n+1} u_{m,k}^{n+1})} \quad (10)$$

3.2.5 Pressure correction

An updated pressure $P_{m,k+1}^{n+1}$ is calculated from:

$$\frac{P_{m,k+1}^{n+1} - P_{m,k}^{n+1}}{(c^2)_{m,k+1/2}^{n+1} \Delta t} - \partial_x \Delta t \partial_x (P_{m,k+1}^{n+1} - P_{m,k}^{n+1}) = -\frac{\rho_{m,k+1/2}^{n+1} - \rho^n}{\Delta t} - \partial_x q_{m,k}^{n+1} \quad (11)$$

with:

$$(c^2)_{m,k+1/2}^{n+1} = \frac{\gamma P_{m,k}^{n+1}}{\rho_{m,k+1/2}^{n+1}} \quad (12)$$

Then, the velocity $u_{m,k+1}^{n+1}$, the mass flux $q_{m,k+1}^{n+1}$ and the density $\rho_{m,k+1}^{n+1}$ are updated:

$$\begin{cases} \frac{u_{m,k+1}^{n+1} - u_{m,k}^{n+1}}{\Delta t} = -\frac{\partial_x (P_{m,k+1}^{n+1} - P_{m,k}^{n+1})}{\rho_{m,k+1/2}^{n+1}} \\ \frac{q_{m,k+1}^{n+1} - q_{m,k}^{n+1}}{\Delta t} = -\partial_x (P_{m,k+1}^{n+1} - P_{m,k}^{n+1}) \end{cases} \quad (13)$$

$$\rho_{m,k+1}^{n+1} = \frac{\gamma P_{m,k+1}^{n+1}}{(\gamma - 1)(H_{m,k+1}^{n+1} - \frac{1}{2} u_{m,k+1}^{n+1} u_{m,k+1}^{n+1})} \quad (14)$$

It may be useful to indicate the origin of the pressure-correction equation (11). We start from the discrete mass equation:

$$\frac{\rho_{m,k+1}^{n+1} - \rho^n}{\Delta t} + \partial_x q_{m,k+1}^{n+1} = 0$$

We replace the mass flux $q_{m,k+1}^{n+1}$ by its expression according to (13):

$$q_{m,k+1}^{n+1} = q_{m,k}^{n+1} - \Delta t \partial_x (P_{m,k+1}^{n+1} - P_{m,k}^{n+1})$$

We split the density variation into two parts as follows:

$$\rho_{m,k+1}^{n+1} - \rho^n = (\rho_{m,k+1}^{n+1} - \rho_{m,k+1/2}^{n+1}) + (\rho_{m,k+1/2}^{n+1} - \rho^n)$$

The first part vanishes when the iterative process converges. It is modelled as follows:

$$\rho_{m,k+1}^{n+1} - \rho_{m,k+1/2}^{n+1} = (P_{m,k+1}^{n+1} - P_{m,k}^{n+1}) / (c^2)_{m,k+1/2}^{n+1}$$

3.2.6 End of the subcycles

The subcycles are not necessarily executed up to $k = nthm$ and $m = ntg$.

To end a given m -subcycle, a convergence test ($C1$) is carried out at the end of the enthalpy step, at each k -iteration (except at the first one). The current m -subcycle is interrupted (and the last pressure step of the subcycle is skipped) if ($C1$) is verified, that is if $k = nthm$ or if the discrete mass equation is satisfied (*i.e.* if the right-hand side of the pressure equation (11) is sufficiently small).

To end a given time step, a convergence test ($C2$) is carried out at the end of each m -subcycle. The current time step is interrupted if ($C2$) is verified, that is, if $m = ntg$ or if the discrete divergence of the mass flux Q_l^{n+1} did not vary between the last two subcycles $l = m - 1$ and $l = m$.

With these two stopping tests, variables are updated differently at the end of the time step and at the end of each m -subcycle:

- at the end of the last m -subcycle of a given time step (the conditions ($C1$) and ($C2$) are true, $m = m_{max} \leq ntg$ and $k = k_{max} \leq nthm$), the variables are updated for the next time-step as follows:

$$(u, P, q)^{n+1} = (u, P, q)_{m_{max}, k_{max}}^{n+1} \quad (15)$$

$$H^{n+1} = H_{m_{max}, k_{max}+1}^{n+1} \quad \text{and} \quad \rho^{n+1} = \rho_{m_{max}, k_{max}+1/2}^{n+1}$$

- at the end of all other m -subcycles (the condition ($C1$) is true, the condition ($C2$) is false, and $k = k_{max} \leq nthm$), only Q needs to be updated for the next m -subcycle:

$$Q_{m+1}^{n+1} = q_{m, k_{max}}^{n+1} \quad (16)$$

3.3 Discretization in space

3.3.1 Velocity prediction

The upwind scheme is used for the convective term $\partial_x(q^n u^*)$, while all other terms are centred. With $u_{c,i+1/2}$ standing for the upwind face value with respect to $q_{i+1/2}^n$ and $P_{i+1/2}^n$ standing for the (centred) linear interpolation for pressure, we have:

$$\begin{aligned} \frac{\rho_i^n \Omega_i}{\Delta t} (u_m^*)_i &= ((Q_m^{n+1})_{i+1/2} - (Q_m^{n+1})_{i-1/2}) S(u_m^*)_i \\ &+ \left(q_{i+1/2}^n (u_m^*)_{c,i+1/2} - q_{i-1/2}^n (u_m^*)_{c,i-1/2} \right) S \\ &= \frac{\rho_i^n \Omega_i}{\Delta t} u_i^n - \left(P_{i+1/2}^n - P_{i-1/2}^n \right) S \end{aligned} \quad (17)$$

Once the velocity field has been predicted, the mass flux is updated, using the so-called Rhie and Chow interpolation, defined below ($\rho_{c,i+1/2}^n$ is the upwind face value):

$$\begin{aligned} (q_m^*)_{i+1/2} &= \frac{1}{2} ((u_m^*)_{i+1} + (u_m^*)_i) \rho_{c,i+1/2}^n \\ &+ \frac{\Delta t}{2} \left(\frac{P_{i+3/2}^n - P_{i+1/2}^n}{\Delta x} + \frac{P_{i+1/2}^n - P_{i-1/2}^n}{\Delta x} \right) - \Delta t \frac{P_{i+1}^n - P_i^n}{\Delta x} \end{aligned} \quad (18)$$

3.3.2 Enthalpy

The upwind scheme is used for the convective term $\partial_x(q_{m,k}^{n+1} H_{m,k+1}^{n+1})$, while all other terms are centred. With $H_{c,i+1/2}$ standing for the upwind face value with respect to $(q_{m,k}^{n+1})_{i+1/2}$, we have:

$$\begin{aligned} \frac{\rho_i^n \Omega_i}{\Delta t} (H_{m,k+1}^{n+1})_i &= \left((q_{m,k}^{n+1})_{i+1/2} - (q_{m,k}^{n+1})_{i-1/2} \right) (H_{m,k+1}^{n+1})_i S \\ &+ \left((q_{m,k}^{n+1})_{i+1/2} (H_{m,k+1}^{n+1})_{c,i+1/2} - (q_{m,k}^{n+1})_{i-1/2} (H_{m,k+1}^{n+1})_{c,i-1/2} \right) S \\ &= \frac{\rho_i^n \Omega_i}{\Delta t} H_i^n + \frac{(P_{m,k}^{n+1})_i - (P^n)_i}{\Delta t} \Omega_i + \Phi_i \Omega_i \end{aligned} \quad (19)$$

3.3.3 Pressure correction

A centred discretization in space is used:

$$\begin{aligned} \frac{\Omega_i}{((c^2)_{m,k+1/2}^{n+1})_i \Delta t} \delta P_i &= \Delta t \left((\partial_x \delta P)_{f,i+1/2} - (\partial_x \delta P)_{f,i-1/2} \right) S \\ &= - \frac{(\rho_{m,k+1/2}^{n+1})_i - \rho_i^n}{\Delta t} \Omega_i - \left((q_{m,k}^{n+1})_{i+1/2} - (q_{m,k}^{n+1})_{i-1/2} \right) S \end{aligned} \quad (20)$$

with the notations:

$$\left\{ \begin{array}{l} \delta P = P_{m,k+1}^{n+1} - P_{m,k}^{n+1} \\ ((c^2)_{m,k+1/2}^{n+1})_i = \frac{\gamma (P_{m,k}^{n+1})_i}{(\rho_{m,k+1/2}^{n+1})_i} \\ (\partial_x \delta P)_{f,i+1/2} = \frac{\delta P_{i+1} - \delta P_i}{\Delta x} \end{array} \right. \quad (21)$$

The left-hand side of the pressure-correction equation shall vanish for k sufficiently large and, in this case, the pressure-correction equation is the discrete form of the continuity equation.

Once the pressure-correction equation has been solved, the mass flux at internal faces $i + 1/2$ (for $0 < i < n_{cel}$) is updated:

$$(q_{m,k+1}^{n+1})_{i+1/2} = (q_{m,k}^{n+1})_{i+1/2} - \Delta t \frac{\delta P_{i+1} - \delta P_i}{\Delta x} \quad (22)$$

Similarly, the velocity is updated. With $\delta P_{i+1/2}$ standing for the linear interpolation of δP at the face $i + 1/2$, we have:

$$(u_{m,k+1}^{n+1})_i = (u_{m,k}^{n+1})_i - \frac{\Delta t}{(\rho_{m,k+1/2}^{n+1})_i} \frac{\delta P_{i+1/2} - \delta P_{i-1/2}}{\Delta x} \quad (23)$$

3.4 Discrete properties

3.4.1 Conservativity

Conservativity in space is ensured by construction with the finite volume scheme used.

Conservativity in time is verified provided the iterative procedures converge (sufficiently large values of ntg and of $nthm$ are required). Indeed, conservativity in time requires:

$$\rho^n \frac{u_{m_{max}}^* - u^n}{\Delta t} - u_{m_{max}}^* \partial_x Q_{m_{max}}^{n+1} = \frac{\rho^{n+1} u^{n+1} - \rho^n u^n}{\Delta t} \quad (24)$$

and:

$$\rho^n \frac{H_{m_{max}, k_{max}+1}^{n+1} - H^n}{\Delta t} - H_{m_{max}, k_{max}+1}^{n+1} \partial_x q_{m_{max}, k_{max}}^{n+1} = \frac{\rho^{n+1} H^{n+1} - \rho^n H^n}{\Delta t} \quad (25)$$

These relations hold if the enthalpy-pressure and the global iterative cycles are converged. Indeed, at convergence we have:

$$u_{m_{max}}^* = u^{n+1} \quad \text{and} \quad H_{m_{max}, k_{max}+1}^{n+1} = H^{n+1} \quad (26)$$

At convergence, we also have:

$$\rho_{m_{max}, k_{max}+1/2}^{n+1} = \rho^{n+1} \quad \text{so that:} \quad \frac{\rho^{n+1} - \rho^n}{\Delta t} + \partial_x q_{m_{max}, k_{max}}^{n+1} = 0 \quad (27)$$

Moreover, convergence yields:

$$\partial_x q_{m_{max}, k_{max}}^{n+1} = \partial_x q_{m_{max}-1, k_{max}}^{n+1} \quad (28)$$

Finally, since:

$$Q_{m_{max}}^{n+1} = q_{m_{max}-1, k_{max}}^{n+1} \quad (29)$$

we have:

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} + \partial_x Q_{m_{max}}^{n+1} = 0 \quad (30)$$

and equations (24) and (25) are recovered.

3.4.2 Preservation of the total enthalpy

For steady-state 1D flows, the total enthalpy H at interfaces is conserved, with the following discrete definition: $H = H_{i+1/2} = H_{c, i+1/2}$. A similar property holds for multidimensional flows.

4 SLK algorithm

4.1 Algorithm overview

A first implementation of the algorithm is described in [18]. The scheme is first order in time. At each time step, the solution proceeds as follows:

- density step: density and mass flux are calculated (the mass flux will be used in the next two steps),
- momentum step: velocity is calculated as a solution of the momentum equation, using the mass flux obtained at the end of the first step (density step),
- energy step: total energy is determined, using the mass flux obtained at the end of the first step (density step) and pressure is updated.

It is pointed out that the so-called Rhie and Chow interpolation is not used in the SLK algorithm.

4.2 Discretization in time

4.2.1 Density step

The density ρ^{n+1} and the mass flux q_{ac}^{n+1} are calculated. Note that q_{ac}^{n+1} will be used for the advective terms in the momentum and energy equations.

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} + \partial_x (q^n - \Delta t \theta \beta \partial_x s^n) - \partial_x (\Delta t (c^2 \partial_x \rho^{n+1})) = 0 \quad (31)$$

$$q_{ac}^{n+1} = q^n - \Delta t (c^2 \partial_x \rho^{n+1} + \theta \beta \partial_x s^n) \quad (32)$$

In these equations and below, θ allows to take $\beta \partial_x s$ into account ($\theta = 1$) or not ($\theta = 0$). Unless otherwise stated, all the computations presented here have been carried out with $\theta = 1$.

4.2.2 Momentum step

The velocity u^{n+1} is calculated as follows:

$$\rho^n \frac{u^{n+1} - u^n}{\Delta t} - u^{n+1} \partial_x q_{ac}^{n+1} + \partial_x (q_{ac}^{n+1} u^{n+1}) = -\partial_x P^n \quad (33)$$

Variants of the algorithm may be derived, using a predicted pressure P^{pred} instead of P^n in the momentum equation (33). For example, using an isothermal approximation:

$$P^{pred} = P^n \frac{\rho^{n+1}}{\rho^n} \quad (34)$$

or, using an isentropic approximation:

$$P^{pred} = P^n \left(\frac{\rho^{n+1}}{\rho^n} \right)^\gamma \quad (35)$$

All the calculations presented here have been carried out using P^n in the momentum equation.

4.2.3 Energy step

The total energy E^{n+1} is calculated:

$$\rho^n \frac{E^{n+1} - E^n}{\Delta t} - E^{n+1} \partial_x q_{ac}^{n+1} + \partial_x \left(q_{ac}^{n+1} \left(E^{n+1} + \frac{P^n}{\rho^{n+1}} \right) \right) = \Phi \quad (36)$$

Finally, the pressure is updated:

$$P^{n+1} = (\gamma - 1) \rho^{n+1} \left(E^{n+1} - \frac{1}{2} u^{n+1} u^{n+1} \right) \quad (37)$$

4.3 Discretization in space

4.3.1 Density step

The upwind scheme is used for the mass flux divergence, while all other terms are centred. With $u_{i+1/2}^n$ standing for the (centred) linear interpolation for velocity and $\rho_{c,i+1/2}$ standing for the upwind face value with respect to $u_{i+1/2}^n$, we have:

$$\begin{aligned} \Omega_i \frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} &+ (\rho_{c,i+1/2}^n u_{i+1/2}^n - \rho_{c,i-1/2}^n u_{i-1/2}^n) S \\ &- \Delta t \left(\theta \beta_{h,i+1/2}^n (\partial_x s^n)_{f,i+1/2} - \theta \beta_{h,i-1/2}^n (\partial_x s^n)_{f,i-1/2} \right) S \\ &- \Delta t \left((c^2)_{h,i+1/2}^n (\partial_x \rho^{n+1})_{f,i+1/2} - (c^2)_{h,i-1/2}^n (\partial_x \rho^{n+1})_{f,i-1/2} \right) S = 0 \end{aligned} \quad (38)$$

In this equation, $\beta_{h,i+1/2}^n$ and $(c^2)_{h,i+1/2}^n$ are harmonic interpolations at the face $i + 1/2$, with the following notation (f standing for β or c^2):

$$f_{h,i+1/2}^n = 2 \frac{f_{i+1}^n f_i^n}{f_{i+1}^n + f_i^n} \quad (39)$$

The gradients $(\partial_x s^n)_{f,i+1/2}$ and $(\partial_x \rho^{n+1})_{f,i+1/2}$ are calculated directly at the face $i + 1/2$ using the values of s and ρ in the two neighbouring cells.

The mass flux at the faces is calculated as follows:

$$\begin{aligned} q_{ac,i+1/2}^{n+1} &= \rho_{c,i+1/2}^n u_{i+1/2}^n \\ &- \Delta t \left(\theta \beta_{h,i+1/2}^n (\partial_x s^n)_{f,i+1/2} + (c^2)_{h,i+1/2}^n (\partial_x \rho^{n+1})_{f,i+1/2} \right) \end{aligned} \quad (40)$$

4.3.2 Momentum step

The upwind scheme is used for the convective term $\partial_x (q_{ac}^{n+1} u^*)$, while all other terms are centred ($u_{c,i+1/2}^{n+1}$ stands for the upwind face value with respect to $q_{ac,i+1/2}^{n+1}$ and $P_{i+1/2}^n$ is the linear interpolation for pressure):

$$\begin{aligned} \frac{\rho_i^n \Omega_i}{\Delta t} u_i^{n+1} &- \left(q_{ac,i+1/2}^{n+1} - q_{ac,i-1/2}^{n+1} \right) S u_i^{n+1} + \left(q_{ac,i+1/2}^{n+1} u_{c,i+1/2}^{n+1} - q_{ac,i-1/2}^{n+1} u_{c,i-1/2}^{n+1} \right) S \\ &= \frac{\rho_i^n \Omega_i}{\Delta t} u_i^n - (P_{i+1/2}^n - P_{i-1/2}^n) S \end{aligned} \quad (41)$$

4.3.3 Energy step

As for momentum, the upwind scheme is used for the convective term $\partial_x \left(q_{ac}^{n+1} \left(E^{n+1} + \frac{P^n}{\rho^{n+1}} \right) \right)$, while all other terms are centred. With $E_{c,i+1/2}^{n+1}$, $P_{c,i+1/2}^n$ and $\rho_{c,i+1/2}^{n+1}$ standing for the upwind face values with respect to $q_{ac,i+1/2}^{n+1}$, we have:

$$\begin{aligned} \frac{\rho_i^n \Omega_i}{\Delta t} E_i^{n+1} &= \left(q_{ac,i+1/2}^{n+1} - q_{ac,i-1/2}^{n+1} \right) S E_i^{n+1} \\ &+ \left(q_{ac,i+1/2}^{n+1} \left(E_{c,i+1/2}^{n+1} + \frac{P_{c,i+1/2}^n}{\rho_{c,i+1/2}^{n+1}} \right) - q_{ac,i-1/2}^{n+1} \left(E_{c,i-1/2}^{n+1} + \frac{P_{c,i-1/2}^n}{\rho_{c,i-1/2}^{n+1}} \right) \right) S \\ &= \frac{\rho_i^n \Omega_i}{\Delta t} E_i^n + \Phi \Omega_i \end{aligned} \quad (42)$$

4.4 Discrete properties

4.4.1 Conservativity

This property is verified by construction of the algorithm.

Conservativity in space is ensured by construction (finite volume scheme). Conservativity in time is ensured if the following relation holds for u (and a similar one for E instead of u):

$$\rho^n \frac{u^{n+1} - u^n}{\Delta t} - u^{n+1} \partial_x q_{ac}^{n+1} = \frac{\rho^{n+1} u^{n+1} - \rho^n u^n}{\Delta t} \quad (43)$$

This relation is ensured by the discrete form of the mass equation (from (31) and (32)):

$$-\partial_x q_{ac}^{n+1} = \frac{\rho^{n+1} - \rho^n}{\Delta t} \quad (44)$$

4.4.2 Positivity of the density

The density equation (38) is recalled hereafter. The convective velocity $w = u - \Delta t \theta \rho^{-1} \beta \partial_x s$ is introduced to deal simultaneously with the cases $\theta = 0$ and $\theta = 1$:

$$\begin{aligned} \frac{\Omega_i}{\Delta t} \rho_i^{n+1} &= \Delta t \left((c^2)_{h,i+1/2}^n (\partial_x \rho^{n+1})_{f,i+1/2} - (c^2)_{h,i-1/2}^n (\partial_x \rho^{n+1})_{f,i-1/2} \right) S \\ &= \frac{\Omega_i}{\Delta t} \rho_i^n - \left(\rho_{c,i+1/2}^n w_{i+1/2}^n - \rho_{c,i-1/2}^n w_{i-1/2}^n \right) S \end{aligned} \quad (45)$$

With the explicit treatment adopted here for the convective term ($\rho_{c,i+1/2}^n w_{i+1/2}^n - \rho_{c,i-1/2}^n w_{i-1/2}^n$), a condition that ensures the discrete positivity of the density for 1D flows may be derived as indicated below (a similar condition holds for multidimensional flows).

The matrix of the system is a M-matrix¹ and it is stricly diagonal dominant². Hence, the positivity of the right-hand side is sufficient to ensure the discrete positivity of the density.

The right-hand side of the density equation reads:

$$b_i = c_{ii} \rho_i^n + c_{i(i+1)} \rho_{i+1}^n + c_{i(i-1)} \rho_{i-1}^n \quad (47)$$

with:

$$\begin{cases} c_{ii} &= \frac{\Omega_i}{\Delta t} - \left(s_p(w_{i+1/2}^n) w_{i+1/2}^n - (1 - s_p(w_{i-1/2}^n)) w_{i-1/2}^n \right) S \\ c_{i(i+1)} &= - \left((1 - s_p(w_{i+1/2}^n)) w_{i+1/2}^n \right) S \\ c_{i(i-1)} &= - \left(-s_p(w_{i-1/2}^n) w_{i-1/2}^n \right) S \end{cases} \quad (48)$$

with $s_p(a)$ returning 1 if a is positive and 0 otherwise.

The coefficients $c_{i(i-1)}$ and $c_{i(i+1)}$ are always positive. Hence, a condition sufficient to ensure the discrete positivity of the density is $c_{ii} \geq 0$, that is:

$$\frac{\Omega_i}{\Delta t} - \left(s_p(w_{i+1/2}^n) w_{i+1/2}^n - (1 - s_p(w_{i-1/2}^n)) w_{i-1/2}^n \right) S \geq 0 \quad (49)$$

or:

$$\frac{\Omega_i}{\Delta t} - \left(\text{Max}(0, w_{i+1/2}^n) - \text{Min}(0, w_{i-1/2}^n) \right) S \geq 0 \quad (50)$$

This condition ensures the discrete positivity of the density. It may also be put under a slightly different form:

$$\frac{\Omega_i}{\Delta t} - \left(\text{Max}(0, w_{i+1/2}^n) + \text{Max}(0, -w_{i-1/2}^n) \right) S \geq 0 \quad (51)$$

For $\theta = 0$ (*i.e.* with $w = u$), the condition (50) may be illustrated as follows, with reference to a CFL condition based on the velocity of the fluid (and not on the speed of sound):

- for a uniform velocity field, the condition corresponds to a “CFL=1” constraint.
- in the centre of a symmetric double rarefaction wave, the condition corresponds to a “CFL=0.5” constraint.
- in the centre of a symmetric double shock wave, the condition does not imply any limitation.

¹A M-matrix is defined as a matrix which has the following properties: all the diagonal terms are non-zero and have the same sign and all the off-diagonal terms have the opposite sign.

²The associated coefficients read:

$$\begin{cases} a_{ii} &= \frac{\Omega_i}{\Delta t} + \frac{\Delta t}{\Delta x} ((c^2)_{h,i+1/2}^n + (c^2)_{h,i-1/2}^n) S \\ a_{i(i+1)} &= -\frac{\Delta t}{\Delta x} (c^2)_{h,i+1/2}^n S \\ a_{i(i-1)} &= -\frac{\Delta t}{\Delta x} (c^2)_{h,i-1/2}^n S \end{cases} \quad (46)$$

With a semi-implicit treatment of the convective term ($\rho_{c,i+1/2}^{n+1} w_{i+1/2}^n - \rho_{c,i-1/2}^{n+1} w_{i-1/2}^n$), a quite different condition is obtained (none of the present computations has been carried out with this option).

The associated density equation reads:

$$\begin{aligned} \frac{\Omega_i}{\Delta t} \rho_i^{n+1} &+ (\rho_{c,i+1/2}^{n+1} w_{i+1/2}^n - \rho_{c,i-1/2}^{n+1} w_{i-1/2}^n) S \\ &- \Delta t \left((c^2)_{h,i+1/2}^n (\partial_x \rho^{n+1})_{f,i+1/2} - (c^2)_{h,i-1/2}^n (\partial_x \rho^{n+1})_{f,i-1/2} \right) S = \frac{\Omega_i}{\Delta t} \rho_i^n \end{aligned} \quad (52)$$

The right-hand side is positive. The positivity of the density is ensured if the matrix of the system is a strictly diagonal dominant M-matrix with positive diagonal terms. The coefficients of the matrix are:

$$\begin{cases} a_{ii} &= \frac{\Omega_i}{\Delta t} + \left(s_p(w_{i+1/2}^n) w_{i+1/2}^n - (1 - s_p(w_{i-1/2}^n)) w_{i-1/2}^n \right) S \\ &\quad + \frac{\Delta t}{\Delta x} \left((c^2)_{h,i+1/2}^n + (c^2)_{h,i-1/2}^n \right) S \\ a_{i(i+1)} &= \left((1 - s_p(w_{i+1/2}^n)) w_{i+1/2}^n \right) S - \frac{\Delta t}{\Delta x} (c^2)_{h,i+1/2}^n S \\ a_{i(i-1)} &= \left(-s_p(w_{i-1/2}^n) w_{i-1/2}^n \right) S - \frac{\Delta t}{\Delta x} (c^2)_{h,i-1/2}^n S \end{cases} \quad (53)$$

From the definition of s_p ($s_p(a)$ returns 1 if a is positive and 0 otherwise) and from the positivity of $(c^2)_{h,i\pm 1/2}^n$, it follows that the off-diagonal coefficients are negative and that the diagonal coefficients are strictly positive.

Hence, we have $|a_{ii}| - |a_{i(i+1)}| - |a_{i(i-1)}| = a_{ii} + a_{i(i+1)} + a_{i(i-1)}$ and a sufficient condition for the positivity of the density is $a_{ii} + a_{i(i+1)} + a_{i(i-1)} > 0$, that is:

$$\frac{\Omega_i}{\Delta t} + (w_{i+1/2}^n - w_{i-1/2}^n) S > 0 \quad (54)$$

This condition (54) may be illustrated as follows, for $\theta = 0$ (*i.e.* with $w = u$):

- for a uniform velocity field or in the centre of a symmetric double rarefaction wave, the condition does not imply any limitation,
- in the centre of a symmetric double shock wave, the condition corresponds to a “CFL=0.5” constraint (CFL based on the velocity of the fluid).

4.4.3 Preservation of the total enthalpy

For steady-state 1D flows, the total enthalpy H at interfaces is conserved, with the following discrete definition: $H = H_{i+1/2} = E_{c,i+1/2} + \frac{P_{c,i+1/2}}{\rho_{c,i+1/2}}$. A similar property holds for multidimensional flows.

5 Approximate Godunov scheme VFRoe-ncv

To provide a reference solution for the present computations with a heat source term, many Godunov-type schemes may be selected, amongst which the most popular are the approximate

Roe scheme [20] and the exact Godunov scheme [13] (see [12] and [23] for a review). In the present study, the approximate Godunov scheme VFRoe-ncv has been adopted. The implementation of the scheme is first order in time.

The VFRoe-ncv scheme was first introduced in [4]. A comparison of this scheme with the exact Godunov scheme in the framework of shallow water equations can also be found in [3]. The ability of VFRoe-ncv to handle very strong rarefaction waves is discussed in [10]. Its ability to calculate moving contact discontinuities is investigated in detail in [11], focusing on Euler equations with complex equations of state.

Moreover, a detailed comparison of VFRoe-ncv to many Riemann solvers is available in [9]. This paper provides rates of convergence for many test cases (including Sod test case, sonic rarefaction waves, ...) and schemes (original VFRoe scheme, Rusanov scheme, VFFC scheme, relaxation schemes and VFRoe-ncv schemes).

The present VFRoe-ncv version uses the non-conservative state-variable $Y = (\rho, u, P)$. As Roe linearized Riemann solver, it requires an entropy fix at sonic points. More details are provided in the appendix.

6 Presentation of the test-cases

6.1 Test-cases without heat source term

For all computations considered in the present section, the following conditions hold.

- The source term Φ is zero and we use $\gamma = 1.4$ for all the test-cases.
- The time interval considered is $[t_{min}; t_{max}]$, with $t_{min} = 0$ and $t_{max} = 3 \cdot 10^{-3} s$, except for the sonic shock tube TST that was run with $t_{max} = 0.2 s$. In space, the domain extends over the interval $[-L/2; L/2]$, with $L = 4 m$. The boundary conditions are zero-flux conditions for all variables. Moreover, the domain is sufficiently large so that no wave reaches the boundaries before the simulations are stopped at $t = t_{max}$.

The initial conditions are detailed hereafter.

Contact discontinuity: CD

$$\begin{cases} \rho &= 1. \text{ for } x < 0 & \text{and} & \rho &= 0.125 \text{ for } x > 0 \\ u &= 200. \\ P &= 10^5 \end{cases} \quad (55)$$

Sod shock tube: SOD

$$\begin{cases} \rho &= 1. \text{ for } x < 0 & \text{and} & \rho &= 0.125 \text{ for } x > 0 \\ u &= 0. \\ P &= 10^5 \text{ for } x < 0 & \text{and} & P &= 10^4 \text{ for } x > 0 \end{cases} \quad (56)$$

Sonic shock tube (Toro [23]): TST

$$\begin{cases} \rho &= 1. \text{ for } x < 0 & \text{and} & \rho &= 0.125 \text{ for } x > 0 \\ u &= 0.75 \text{ for } x < 0 & \text{and} & u &= 0 \text{ for } x > 0 \\ P &= 1 \text{ for } x < 0 & \text{and} & P &= 0.1 \text{ for } x > 0 \end{cases} \quad (57)$$

Symmetric double expansion (or “double expansion wave”): SDE

$$\begin{cases} \rho &= 1. \\ u &= -100. \text{ for } x < 0 \quad \text{and} \quad u = 100. \text{ for } x > 0 \\ P &= 10^5 \end{cases} \quad (58)$$

Symmetric double shock (or “double shock wave”): SDS

$$\begin{cases} \rho &= 1. \\ u &= 100. \text{ for } x < 0 \quad \text{and} \quad u = -100. \text{ for } x > 0 \\ P &= 10^5 \end{cases} \quad (59)$$

The method used to calculate the analytical solutions is presented in [22].

6.2 Analytical case with heat source term (case A)

With $\xi = \frac{x}{t}$ and A an arbitrary constant, the heat source term Φ considered here is:

$$\Phi = -\frac{A}{27} \frac{2-\gamma}{\gamma-1} \frac{1}{t} \xi^4 \quad (60)$$

Indeed, with this source term, an analytical solution exists:

$$\begin{cases} \rho &= A \xi^2 \\ u &= \frac{2}{3} \xi \\ P &= \frac{A}{18} \xi^4 \end{cases} \quad (61)$$

The length of the domain on which errors are calculated is the same as for the above test-cases without heat source term ($L = 4 \text{ m}$). Nevertheless, to make the influence of the (Dirichlet) boundary conditions as small as possible, the computational domain is extended on both sides. Hence, the computational domain extends over the interval $[x_{min}; x_{max}]$ with $x_{min} = 1$ and $x_{max} = x_{min} + 3L = 13$ (so that $x > 0$) and the domain on which errors are calculated is restricted to the following interval: $[x_{min} + L; x_{max} - L]$, that is [5; 9].

The time interval considered is $[t_{min}; t_{max}]$, with $t_{min} = 0.010 \text{ s}$ and $t_{max} = 0.011 \text{ s}$ (so that $t > 0$).

To keep the variables between reasonable limits, the constant value $A = 10^{-5}$ has been adopted.

6.3 Test-case with a constant heat source term (case B)

The initial conditions are the following:

$$\begin{cases} \rho &= 1. \\ u &= 0. \\ P &= 10^5 \end{cases} \quad (62)$$

The source term Φ is constant and defined as follows:

$$\begin{aligned} \Phi(x, t > 0) &= 10^{10} \quad \text{for} \quad |x| < \frac{L_\Phi}{2} = 0.1 \\ &= 0 \quad \text{elsewhere} \end{aligned} \quad (63)$$

The boundary conditions are zero-flux conditions for all variables. Moreover, the domain is sufficiently large ($L = 4m$) so that no wave reaches the boundaries before simulations are stopped at $t = t_{max} = 1.4 \cdot 10^{-3} s$.

The reference solution considered has been obtained using VFRoe-ncv method [4], with an automatic limitation of the time step and on a mesh 27 times as fine as the finest mesh used for the pressure-correction solvers (*i.e.* on a mesh containing 174,960 regular cells).

7 Results

7.1 Time step management

All schemes considered in the present paper are first order in time (and space).

The computations with VFRoe-ncv were carried out with an automatic time step limitation: the time step is uniform in space and varies in time to keep the CFL number lower or equal to 0.5 ($(u + c) \Delta t / \Delta x \leq 0.5$).

For each computation with NLK and SLK, the time step is uniform in space and constant in time³. For a given test case, convergence studies have been carried out keeping $\Delta t / \Delta x$ constant for all meshes. The maximum values of the CFL numbers obtained at the end of each calculation are provided in the following sections.

7.2 Error calculation

On the uniform meshes considered here, f_i standing for the value of a variable f at the centre of cell i , the “ L^1 -error” or “ L^1 -difference” between f and a reference solution f_{ref} is evaluated as follows, unless otherwise stated:

$$e_{L^1}^f = \frac{\sum_{i=1}^{ncel} |f_i - f_{ref,i}|}{\sum_{i=1}^{ncel} |f_{ref,i}|} \quad (64)$$

For the analytical case with a heat source term, the summations are not carried out over the whole domain, but only over the cells located in the interval of interest $[x_{min} + L; x_{max} - L]$ defined in section 6.2.

The orders of convergence is estimated from the L^1 -errors (or L^1 -difference) obtained on the finest two meshes.

7.3 Test-cases without heat source term

For the tests reported in the present section, NLK algorithm is operated using large values for the maximum numbers of subcycles: $ntg = 20$ and $nthm = 20$; tests have also been carried out with smaller values of these numbers, to study the effect of the potential loss of conservativity in time on the convergence towards the exact solution.

³This choice has been motivated by the need for comparing easily the transients during the early tests. Moreover, the aim of the present study was not to make any assessment related to the CPU requirements of the algorithms but to show they had the capability of addressing shock solutions.

Convergence is studied as the mesh is refined (80, 160, 320, 640, 1,280, 2,560, 5,120 and 10,240-cell meshes) keeping the ratio $\frac{\Delta t}{\Delta x}$ constant for all meshes. The time step value on the coarser mesh is $\Delta t = 2.10^{-5} s$ for all cases but TST for which we use $\Delta t = 8.10^{-3} s$. The maximum values of the CFL numbers obtained at the end of the computations are summarized in table 2.

	CD	SOD	TST	SDE	SDS
$u\Delta t/\Delta x$	0.08	0.12	0.23	0.04	0.04
$(u+c)\Delta t/\Delta x$	0.50	0.28	0.45	0.19	0.19

Table 2: Maximum CFL numbers obtained at the end of the calculation of the test-cases without heat source term.

Using Smoller’s method [22] to calculate the reference solution for the cases described in section 6.1, the L^1 -norm of the error for density, velocity, pressure and total energy is plotted in figures 4 to 7 and the corresponding values of the orders of convergence obtained from the L^1 -error on the finest two meshes are indicated in tables 3 to 6.

For these homogeneous test-cases, we expect to retrieve the following classical results for *so-called "first-order" schemes*.

- When considering a concentration C whose variation only occurs through the contact discontinuity, we expect to find at time t_0 that $e_{L^1}^C(\Delta x, t_0) = B_C(t_0)\Delta x^{1/2}$ (where $B_C(t_0)$ is a constant depending only on t_0). If we turn to the velocity and pressure variables, one should observe that $e_{L^1}^{U,P}(\Delta x, t_0) = B_{U,P}(t_0)\Delta x$, due to the fact that these variables remain unchanged across the linearly degenerated field.
- If we focus on the density and total energy variables, which vary through both the contact discontinuity and the genuinely non linear fields, the competition between both rates of convergence should result in the following behaviour on very fine meshes: $e_{L^1}^{\rho,E}(\Delta x, t_0) = B_{\rho,E}(t_0)\Delta x^{1/2}$. Nonetheless, if one restricts to intermediate mesh sizes, the rate of convergence should be an average value in the range $[\frac{1}{2}, 1]$.

The analysis of the results reported hereafter can be summarized as follows: **these test-cases clearly demonstrate that it is possible to capture shocks, expansions and contact discontinuities with a conservative pressure-correction algorithm.** Details are provided below.

- For the double shock wave and Sod shock tube (which both involve shocks), the NLK algorithm converges towards the correct solution as the mesh is refined when the maximum numbers of subcycles are sufficiently large (here $ntg = 20$ and $nthm = 20$): see figure 5 and table 4. If the maximum numbers of subcycles are too small (typically, $ntg = 1$ and $nthm = 2$), the algorithm NLK no longer converges towards the right shock solution (figure 7 and tables 5 and 6). This shortcoming is also visible with the choice ($ntg = 1$ and $nthm = 20$). This can be easily explained by the fact that the discrete time derivative of the mean momentum is not strictly conservative unless convergence (with respect to m -subcycles associated with ntg) is achieved. This is a well known feature for non-conservative schemes which are used for the computation of systems of conservation laws (see [15] and [11] for instance). The SLK algorithm, which preserves the conservative form by construction, does not encounter this difficulty.

SLK	CD	SOD	TST	SDE	SDS
density	0.501	0.619	0.611	0.864	0.996
velocity	0.987	0.844	0.778	0.874	1.011
pressure	0.993	0.863	0.771	0.864	0.999

Table 3: Orders of convergence for the test-cases without heat source term: SLK algorithm

NLK(20,20)	CD	SOD	TST	SDE	SDS
density	0.500	0.592	0.568	0.805	1.093
velocity	1.002	0.585	0.525	0.809	1.082
pressure	0.999	0.704	0.592	0.805	1.104

Table 4: Orders of convergence for the test-cases without heat source term: NLK algorithm with $ntg = 20$ and $nthm = 20$

NLK(01,20)	CD	SOD	TST	SDE	SDS
density	0.500	0.188	0.215	0.792	0.125
velocity	1.001	0.053	0.055	0.795	0.258
pressure	0.989	0.101	0.124	0.796	0.131

Table 5: Orders of convergence for the test-cases without heat source term: NLK algorithm with $ntg = 1$ and $nthm = 20$

NLK(01,02)	CD	SOD	TST	SDE	SDS
density	0.500	0.214	0.222	0.795	0.118
velocity	1.002	0.053	0.056	0.800	0.174
pressure	0.989	0.108	0.136	0.798	0.124

Table 6: Orders of convergence for the test-cases without heat source term: NLK algorithm with $ntg = 1$ and $nthm = 2$

- If we restrict to the test-cases involving a pure contact discontinuity or a double rarefaction wave, we retrieve a convergence rate between 0.5 and 1 for ρ and E , whatever the algorithm (and whatever the numbers of subcycles for NLK). Similarly, the rate of convergence is close to 1 for U and P for both SLK and NLK: see figures 4, 5, 6 and 7, and tables 3, 4, 5 and 6.
- If one turns to the approximate Godunov scheme VFRoe-ncv, the reader is referred to [9] for measures of error in L^1 -norm for the same test-cases. We emphasize that the rates of convergence are very similar to those measured here for the SLK algorithm.

Finally, we underline that neither NLK nor SLK require any specific treatment for the sonic shock tube at $x = 0$ (see figure 1). Moreover, we also checked that the SLK algorithm converges towards the exact solution for a very strong rarefaction case (shock tube with $\gamma = 1.01$, starting with the following initial state: $(\rho, u, P) = (10^3, 0., 10^5)$ for $x < 0$ and $(\rho, u, P) = (10^{-2}, 0., 5)$ for $x \geq 0$).

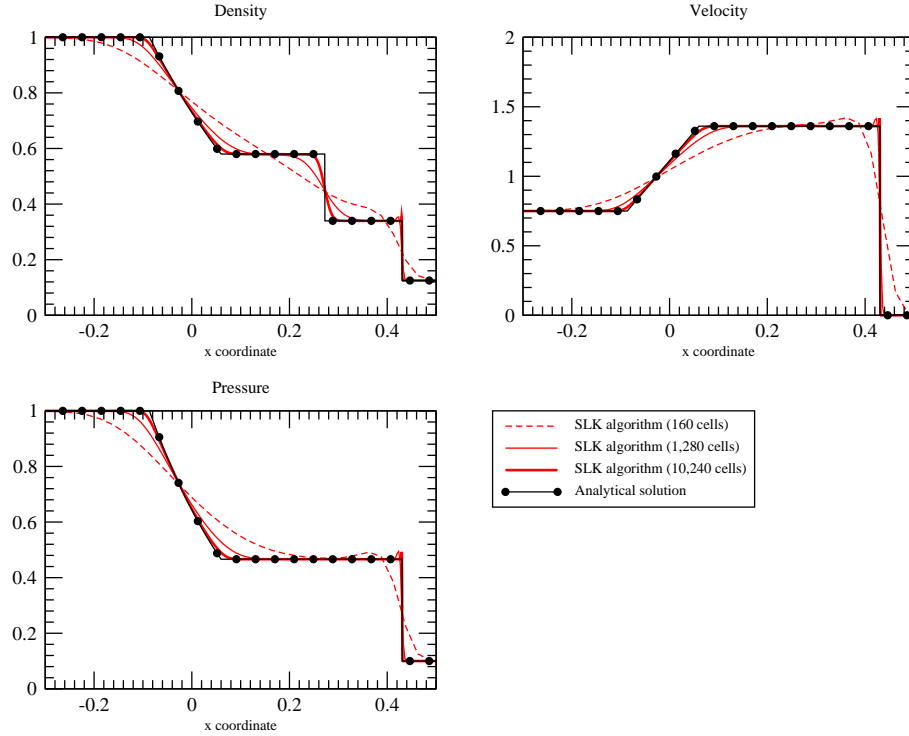


Figure 1: Sonic shock tube. Comparison of the density, velocity and pressure profiles obtained with the SLK algorithm (160, 1,280 and 10,240-cell meshes) with the analytical solution at $t = t_{max} = 0.2$ s

7.4 Analytical case with a heat source term (case A)

This case corresponds to the source term (60) and to the solution (61). It provides the opportunity of studying the convergence of the algorithm as the mesh is refined on a regular solution with a heat source term.

For this test-case, NLK algorithm is operated with the default values of the subcycle parameters: $ntg = 1$ and $nthm = 2$.

The mesh refinement and the time step are the same as those used for the test-cases without heat source term. Convergence is studied as the mesh is refined keeping the ratio $\frac{\Delta t}{\Delta x}$ constant for all meshes, with $\Delta t = 2 \cdot 10^{-5} s$ on the coarsest mesh. The maximum values of the CFL numbers obtained at the end of the computations are $u\Delta t/\Delta x = 0.32$ and $(u + c)\Delta t/\Delta x = 0.45$.

Both pressure-correction algorithms converge satisfactorily (figure 8 and table 7): first order rate of convergence is achieved on all quantities, as expected, since the solution is regular (at least C^1).

	SLK	NLK(01,02)
density	1.000	1.000
velocity	0.999	0.999
pressure	0.999	0.999

Table 7: *Orders of convergence for case A*

7.5 Test-case with constant heat source term (case B)

This case corresponds to the initial condition (62) and to the source term (63).

The convergence of the pressure-correction solvers is studied as the mesh is refined (80, 240, 720, 2,160 and 6,480-cell meshes⁴), keeping a constant ratio $\frac{\Delta t}{\Delta x}$, with $\Delta t = 2 \cdot 10^{-5} s$ on the coarsest mesh for NLK, and with $\Delta t = 2 \cdot 10^{-7} s$ on the coarsest mesh for SLK (SLK fails with $\Delta t = 2 \cdot 10^{-5} s$).

The maximum values of the CFL numbers obtained at the end of the computations with SLK are: $u\Delta t/\Delta x = 1.6 \cdot 10^{-3}$ and $(u + c)\Delta t/\Delta x = 0.49$ (the corresponding values obtained for NLK are approximately 100 times larger).

The solution provided by VFRoe-ncv on a mesh containing $80 \times 3^7 = 174,960$ cells has been taken as the reference solution.

For this test-case, the NLK algorithm has first been operated with the default values of the subcycle parameters, $ntg = 1$ and $nthm = 2$, since satisfactory results for the previous test-case (case A) had been obtained with these values. Then, complementary computations have been carried out with larger values of the subcycle parameters, $ntg = 20$ and $nthm = 20$, to study the effect of ensuring exact conservativity.

⁴On the 80-cell mesh, the heating zone extends over only 2 cells. The meshes are refined successively by a factor of 3 so that transferring the VFRoe-ncv reference solution to the cell centres of coarser meshes does not require any interpolation.

Some solutions obtained with NLK and SLK are plotted in the figure 2 along with the reference solution obtained with VFRoe-ncv. A zoom on the profiles close to the border of the heating zone is presented in the figure 3.

When the mesh is refined, the solutions obtained with NLK (with enough sub-iterations) and with SLK converge towards the VFRoe-ncv reference solution. Indeed, from the table 8 and from the figure 9, we can conclude that:

- the solutions provided by NLK operated with ($ntg = 20$ and $nthm = 20$) and by SLK converge towards the VFRoe-ncv reference solution when the mesh is refined (it is pointed out that, in this case, both NLK and SLK are fully conservative),
- if NLK is operated with $ntg = 1$ and $nthm = 2$, the solution does not converge towards the reference solution obtained with VFRoe-ncv as the mesh is refined.

A difficulty is encountered by NLK and SLK at the border of the heating zone (figure 3):

- on coarse meshes⁵ (plots are not presented here), the profile obtained with SLK exhibits large velocity oscillations in space, and NLK operated with $ntg = 20$ and $nthm = 20$ produces an even more unphysical solution. The velocity profile produced by NLK with $ntg = 1$ and $nthm = 2$ does not exhibit such oscillations.
- as the mesh is refined, the spatial oscillations observed on the solutions produced by NLK and SLK tend to diminish. On finer meshes, the velocity solution obtained with NLK is less perturbed than that obtained with SLK.

	SLK	NLK(01,02)	NLK(20,20)
density	0.724	0.265	0.609
velocity	1.077	0.050	1.014
pressure	1.033	0.005	0.881

Table 8: Orders of convergence for case B (for NLK, the orders of convergence have been calculated using the errors obtained on the 2,160 and the 720-cell meshes)

8 Conclusion

The ability to solve compressible flows with pressure-correction schemes has been investigated. Two algorithms have been studied.

The SLK algorithm sequentially solves mass, momentum and energy without sub-iteration. It is conservative. Under classical conditions, it ensures the discrete positivity of the density and of the scalar variables and preserves the total enthalpy.

The NLK algorithm sequentially solves momentum, enthalpy and pressure with sub-iterations which ensure conservativity. Under classical conditions, it preserves the total enthalpy.

⁵It is important to observe the behaviour of the solution on coarse meshes in the perspective of three-dimensional computations. With the 240-cell mesh, the heating zone extends over 6 cells.

The present conservative pressure-correction algorithms capture relevant shock solutions.

Indeed, both algorithms exhibit satisfactory convergence properties on standard compressible transients (double shock wave, double rarefaction wave, contact discontinuity, Sod shock tube, sonic shock tube) and on a regular test-case with a heat source term. Moreover, it is demonstrated numerically that if the maximum numbers of iterations are not sufficiently large to ensure conservativity, the NLK algorithm does not capture the right shock solutions. The SLK algorithm is not subjected to this difficulty since it is conservative by nature and does not involve any iterative process.

At last, a stiff test-case with a constant heat source term has been studied. When the mesh is refined, SLK and NLK solutions converge towards the VFRoe-ncv solution chosen as a reference solution (NLK must be operated with sufficiently large maximum numbers of sub-iterations). Both algorithms SLK and NLK exhibit stable velocity oscillations at the border of the heating zone; these oscillations diminish as the mesh is refined. On sufficiently fine meshes, the oscillations are smaller with the NLK algorithm than they are with the SLK algorithm. On coarse meshes, the NLK algorithm exhibits a more satisfactory behaviour (with respect to the velocity profile) if it is operated with a small number of sub-iterations (but, consequently, with a loss of conservativity).

Acknowledgments

Part of the work presented here has been achieved in the framework of the NEPTUNE project [14], with financial support from CEA (Commissariat à l'Énergie Atomique), EDF, IRSN (Institut de Radioprotection et de Sécurité Nucléaire) and AREVA-NP.

The authors are grateful to NEPTUNE.CFD [19] and *Code_Saturne* [1] development teams for their support with the algorithms implemented in these codes. The authors also wish to thank C. Delalandre for providing the description of the physical electric arc problem and the initial 3D simulations that motivated the present study. Finally, the authors wish to thank M. Ourau for his participation into these 3D simulations.

References

- [1] Archambeau F, Méchitoua N, Sakiz M. *Code_Saturne*: A finite volume code for the computation of turbulent incompressible flows - Industrial applications. International Journal on Finite Volumes, Vol. 1, 2004, <http://averoes.math.univ-paris13.fr/>
- [2] Bijl H, Wesseling P. A unified method for computing incompressible and compressible flows in boundary-fitted coordinates. Journal of Computational Physics, vol. 141 (2), pp. 153-173, 1998.
- [3] Buffard T, Gallouët T, Hérard JM. A naive scheme to solve shallow-water equations. CRAS Paris, I-326, pp. 385-390, 1998.
- [4] Buffard T, Gallouët T, Hérard JM. A sequel to a rough Godunov scheme: application to real gases. Computers and Fluids, vol. 29 (7), pp. 813-847, 2000.
- [5] Dafermos CM. Hyperbolic conservation laws in continuum physics. Berlin Heidelberg, Springer-Verlag, 2000.

- [6] Delalondre C. Exemples de modélisation d'arcs électriques 3D instationnaires pour des applications industrielles. Colloque CAE, Orléans, France, 14-15 march 2005, in French.
- [7] Douce A, Delalondre C, Biauasser H, Guillot JB. Numerical modelling of an anodic metal bath heated with an Argon transferred arc. The International Iron and Steel Institute of Japan, vol. 43 (8), pp. 1128-1135, 2003.
- [8] Edwards JR, Liou MS. Low-diffusion flux-splitting methods for flows at all speed. AIAA Journal, vol. 36 (9), pp. 1610-1617, 1998.
- [9] Gallouët T, Hérard JM, Seguin N. Some recent finite volume schemes to compute Euler equations using real gas EOS. International Journal for Numerical Methods in Fluids, vol. 39 (12), pp. 1073-1138, 2002.
- [10] Gallouët T, Hérard JM, Seguin N. On the use of symetrizing variables for vacuum. CAL-COLO, vol. 40 (3), pp. 163-194, 2003.
- [11] Gallouët T, Hérard JM, Seguin N. A hybrid scheme to compute contact discontinuities in one dimensional Euler systems. Mathematical Modelling and Numerical Analysis, vol. 36 (6), pp. 1133-1159, 2002.
- [12] Godlewski E, Raviart PA. Numerical approximation of hyperbolic systems of conservation laws. New-York, Springer-Verlag, 1996.
- [13] Godunov SK. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. Mat. Sb., vol. 47, pp. 271-300, 1959.
- [14] Guelfi A, Bestion D, Boucker M, Boudier P, Fillion P, Grandotto M, Hérard JM, Hervieu É, Péturaud P. NEPTUNE - A new software platform for advanced nuclear thermal hydraulics. Nuclear Science and Engineering, vol. 156, pp. 281-324, 2007.
- [15] Hou X, Le Floch PG. Why non conservative schemes converge to wrong solutions: error analysis. Mathematics of Computation, vol. 62, pp. 497-530, 1994.
- [16] Nerinckx K, Vierendeels J, Dick E. Mach-uniformity through the coupled pressure and temperature correction algorithm. Journal of Computational Physics, vol. 206, pp. 597-623, 2005.
- [17] LeVeque RJ. Numerical methods for conservation laws. Basel, Birkhäuser-Verlag, 1990.
- [18] Mathon P, Archambeau F, Hérard JM. Implantation d'un algorithme compressible dans *Code_Saturne*. EDF internal report HI83-03/016, in French, 2003.
- [19] Méchitoua N, Boucker M, Laviéville J, Hérard JM, Pigny S, Serre G. An unstructured finite volume solver for two-phase water/vapour flows modelling based on an elliptic oriented fractional step method. NURETH 10, 10th International Meeting on Nuclear Reactor, Thermal-Hydraulics, Seoul, South Korea, October 5-9, 2003.
- [20] Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. Journal of Computational Physics, vol. 43, pp. 357-372, 1981.
- [21] Schneider T, Botta N, Geratz KJ, Klein R. Extension of finite volume compressible flow solvers to multi-dimensional, variable density zero Mach number flows. Journal of Computational Physics, vol. 155, pp. 248-286, 1999.
- [22] Smoller J. Shock waves and reaction-diffusion equations. New-York, Springer-Verlag, 1983.

- [23] Toro E. Riemann solvers and numerical methods for fluid dynamics. Berlin, Springer-Verlag, 1999.
- [24] Van Der Heul DR, Vuik C, Wesseling P. A conservative pressure-correction method for flow at all speeds. *Computers and Fluids*, vol. 32, pp. 1113-1132, 2003.
- [25] Vierendeels J, Rienslagh K, Dick E. A multigrid semi-implicit line method for viscous incompressible and low-Mach-number flows on high aspect ratio grids. *Journal of Computational Physics*, vol. 154 (2), pp. 310-341, 1999.

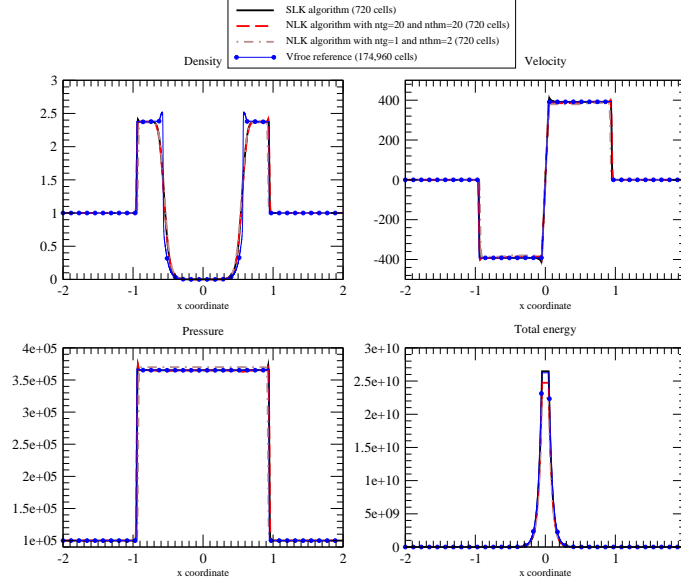


Figure 2: Comparison of the density, velocity, pressure and total energy profiles obtained with the SLK, NLK (720-cell mesh) and VFRoe-ncv (174,960-cell mesh) algorithms on the constant heat source term case (case B) at $t = t_{max} = 1.4 \cdot 10^{-3} s$.

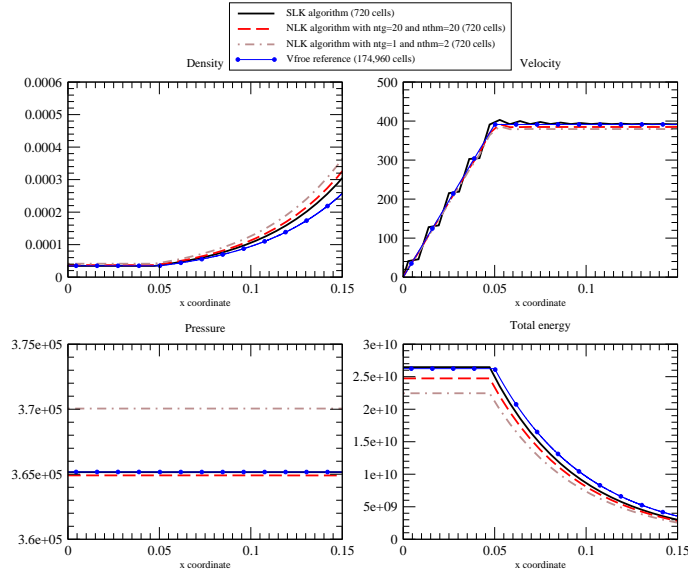


Figure 3: Comparison of the density, velocity, pressure and total energy profiles obtained with the SLK, NLK (720-cell mesh) and VFRoe-ncv (174,960-cell mesh) algorithms on the constant heat source term case (case B) at $t = t_{max} = 1.4 \cdot 10^{-3} s$ - Detailed view close to the border of the heat source zone.

Appendix

Approximate Godunov scheme VFRoe-ncv

To provide a reference solution for the present computations with a heat source term, many Godunov-type schemes may be selected, amongst which the most popular are the approximate Roe scheme [20] and the exact Godunov scheme [13] (see [12] and [23] for a review). In the present study, the VFRoe-ncv [4] scheme has been adopted.

We recall herein the basis of the VFRoe-ncv scheme, which has also been investigated and compared with various Riemann solvers in references [9] and [11]. For the sake of simplicity, we restrict to regular meshes of size Δx such that: $\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$, $i \in \mathbb{Z}$. The time step is $\Delta t = t^{n+1} - t^n$, $n \in \mathbb{N}$. We define $W : \mathbb{R} \times \mathbb{R}^+ \rightarrow \mathbb{R}^p$ the exact solution of the hyperbolic system:

$$\begin{cases} \frac{\partial W}{\partial t} + \frac{\partial F(W)}{\partial x} = 0 \\ W(x, 0) = W_0(x) \end{cases}$$

where the spatial flux F is $C^1(\mathbb{R}^p, \mathbb{R}^p)$. Let W_i^n be the approximate value of $\frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} W(x, t^n) dx$.

Integrating over $[x_{i-\frac{1}{2}}; x_{i+\frac{1}{2}}] \times [t^n; t^{n+1}]$ provides:

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x} \left(\phi_{i+\frac{1}{2}}^n - \phi_{i-\frac{1}{2}}^n \right)$$

where $\phi_{i+\frac{1}{2}}^n$ stands for the numerical flux through the interface $\{x_{i+\frac{1}{2}}\} \times [t^n; t^{n+1}]$. The time step is in agreement with some CFL condition in order to gain stability. $\phi_{i+\frac{1}{2}}^n$ usually only depends on W_i^n and W_{i+1}^n when restricting to so-called first order schemes, that is $\phi_{i+\frac{1}{2}}^n = \phi(W_i^n, W_{i+1}^n)$. Whatever the scheme is, the numerical flux must comply with the consistent condition (see [17]):

$$\phi(V, V) = F(V)$$

We present now approximate Godunov fluxes $\phi(W_L, W_R)$ associated with the 1D Riemann problem:

$$\begin{cases} \frac{\partial W}{\partial t} + \frac{\partial F(W)}{\partial x} = 0 \\ W(x, 0) = \begin{cases} W_L & \text{if } x < 0 \\ W_R & \text{otherwise} \end{cases} \end{cases} \quad (65)$$

VFRoe-ncv scheme is an approximate Godunov scheme where the approximate value at the interface between two cells is calculated as detailed below. Let us consider some change of variable $Y = Y(W)$ in such a way that $W_Y(Y)$ is invertible. The counterpart of above system for regular solutions is:

$$\frac{\partial Y}{\partial t} + B(Y) \frac{\partial Y}{\partial x} = 0$$

where $B(Y) = (W_Y(Y))^{-1} A(W(Y))$ ($W_Y(Y)$ ($A(W)$ stands for the Jacobian matrix of flux $F(W)$)).

Now, the numerical flux $\phi(W_L, W_R)$ is obtained by solving the linearized hyperbolic system:

$$\begin{cases} \frac{\partial Y}{\partial t} + B(\hat{Y}) \frac{\partial Y}{\partial x} = 0 \\ Y(x, 0) = \begin{cases} Y_L = Y(W_L) & \text{if } x < 0 \\ Y_R = Y(W_R) & \text{otherwise} \end{cases} \end{cases} \quad (66)$$

where \hat{Y} agrees with the condition: $\hat{Y}(Y_L, Y_L) = Y_L$, and also $\hat{Y}(Y_L, Y_R) = \hat{Y}(Y_R, Y_L)$.

The classical choice for \hat{Y} in (66) is :

$$\hat{Y}(Y_L, Y_R) = \hat{Y}^{(1)}(Y_L, Y_R) = (Y_L + Y_R)/2$$

Hence it enables us to calculate $Y^*(\frac{x}{t}; Y_L, Y_R)$, and thus to set:

$$\phi(W_L, W_R) = F(W(Y^*(0; Y_L, Y_R)))$$

In order to calculate $Y^*(\frac{x}{t}; Y_L, Y_R)$, we need to introduce \tilde{l}_k , $\tilde{\lambda}_k$ and \tilde{r}_k , $k = 1, \dots, p$, left eigenvectors, eigenvalues and right eigenvectors of matrix $B(\hat{Y})$ respectively. The solution $Y^*(\frac{x}{t}; Y_L, Y_R)$ of the linear Riemann problem (66) is defined everywhere (except along $\frac{x}{t} = \tilde{\lambda}_k$):

$$\begin{aligned} Y^*\left(\frac{x}{t}; Y_L, Y_R\right) &= Y_L + \sum_{\frac{x}{t} > \tilde{\lambda}_k} ({}^t\tilde{l}_k \cdot (Y_R - Y_L)) \tilde{r}_k \\ &= Y_R - \sum_{\frac{x}{t} < \tilde{\lambda}_k} ({}^t\tilde{l}_k \cdot (Y_R - Y_L)) \tilde{r}_k \end{aligned}$$

Combining the last equalities enables us to write the latter in a slightly different form:

$$Y_R - Y_L = \sum_{k=1, p} ({}^t\tilde{l}_k \cdot (Y_R - Y_L)) \tilde{r}_k = \sum_{k=1, p} \tilde{\alpha}_k \tilde{r}_k \quad (67)$$

setting: $\tilde{\alpha}_k = {}^t\tilde{l}_k \cdot (Y_R - Y_L)$.

The explicit form of the Finite Volume method VFRoe-ncv is:

$$W_i^{n+1} - W_i^n + \frac{\Delta t}{\Delta x} (F(W(Y^*(0; Y_i^n, Y_{i+1}^n))) - F(W(Y^*(0; Y_{i-1}^n, Y_i^n)))) = 0$$

Convergence plots for the test-cases without heat source term

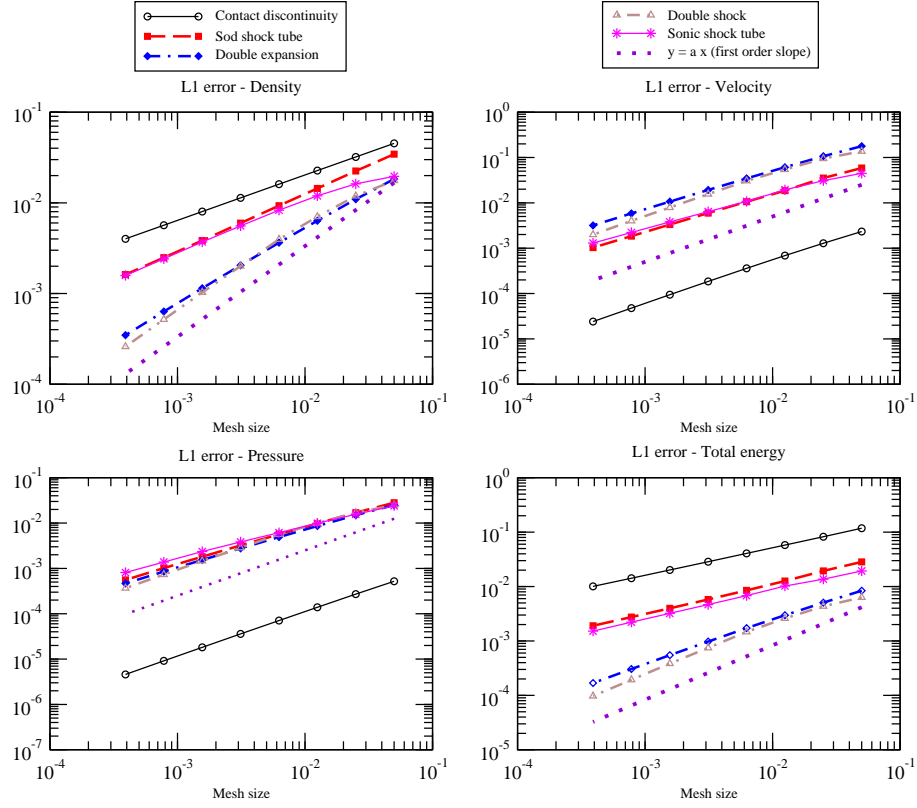


Figure 4: *SLK algorithm - L^1 -errors for the test-cases without heat source term.*

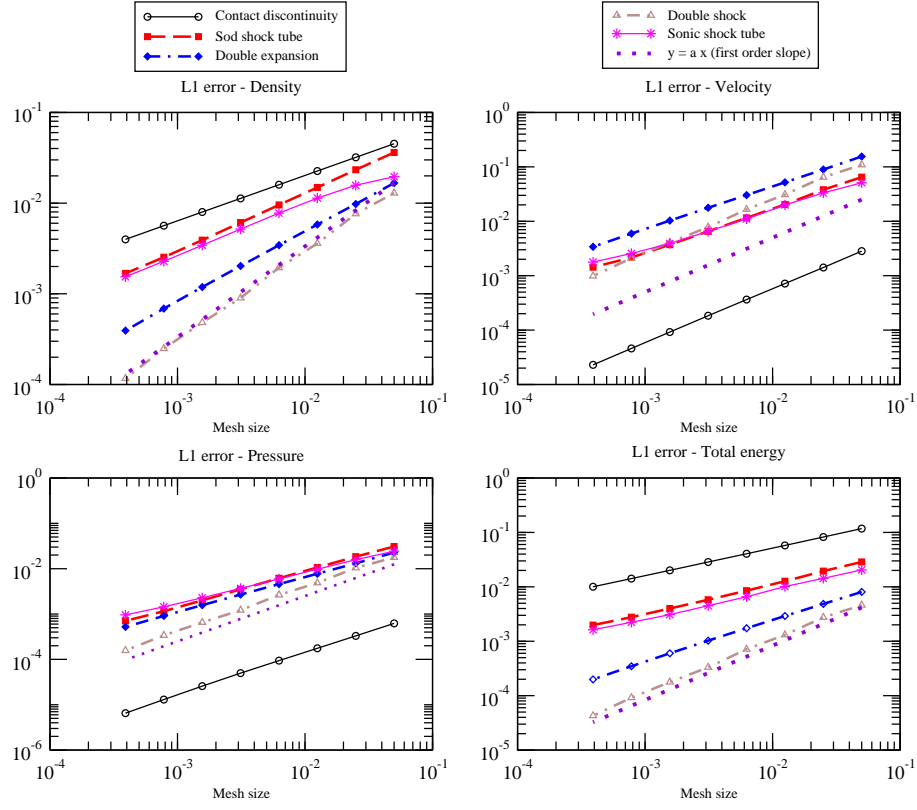


Figure 5: *NLK* algorithm with $ntg = 20$ and $nthm = 20$ - L^1 -errors for the test-cases without heat source term.

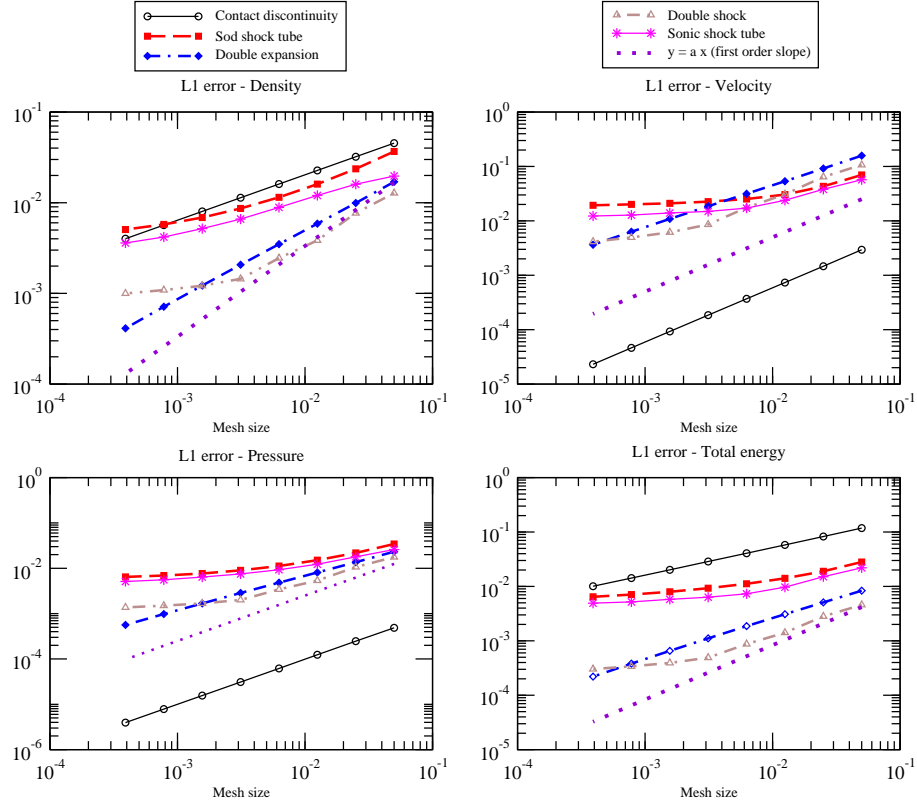


Figure 6: *NLK algorithm with $ntg = 1$ and $nthm = 20$ - L^1 -errors for the test-cases without heat source term.*

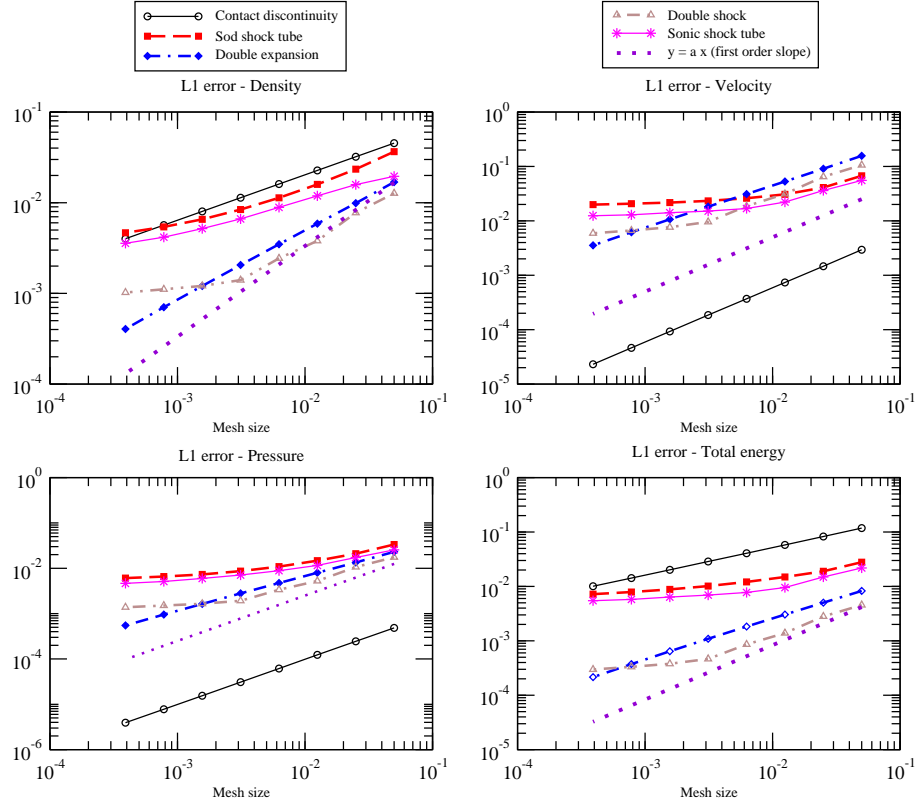


Figure 7: *NLK* algorithm with $ntg = 1$ and $nthm = 2$ - L^1 -errors for the test-cases without heat source term.

Convergence plots for the analytical test-case with a heat source term

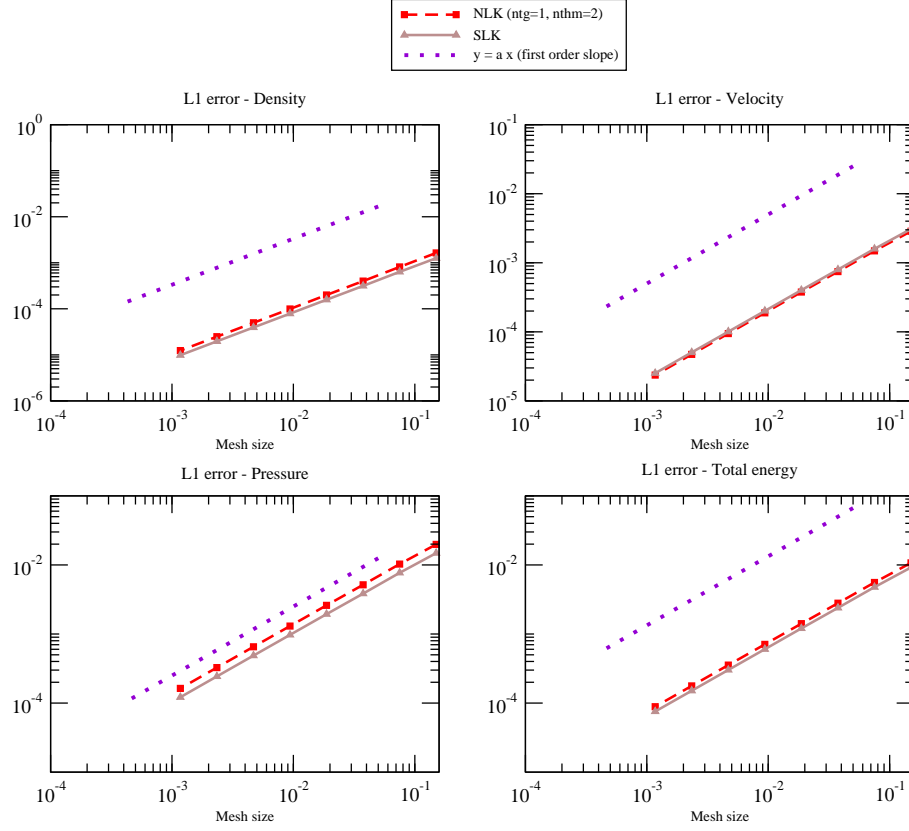


Figure 8: *NLK* and *SLK* L^1 -errors for the analytical test-case with a heat source term (case A); the results with *NLK* were obtained using $ntg = 1$ and $nthm = 2$.

Convergence plots for the test-case with a constant heat source term

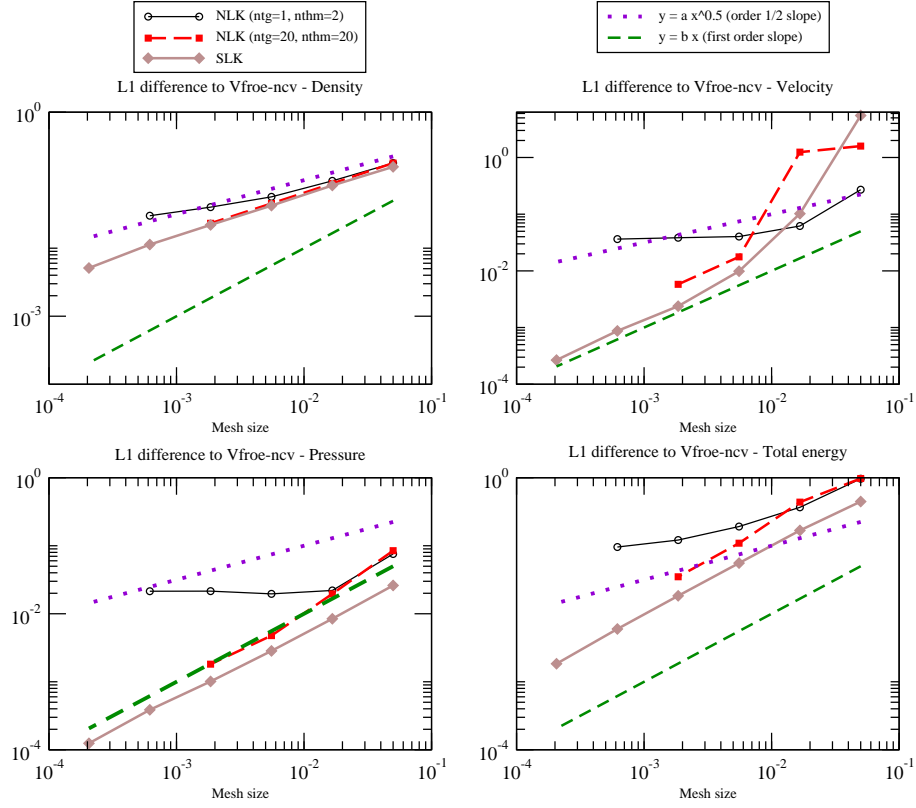


Figure 9: L^1 -difference between the NLK and VFRoe-ncv solutions (curves NLK) and between the SLK and VFRoe-ncv solutions (curves SLK) for the test-case with a constant heat source term (case B); the results with NLK were obtained using two sets of parameters: ($ntg = 1$ and $nthm = 2$) and ($ntg = 20$ and $nthm = 20$). The dotted and the dashed lines help to visualize the convergence slopes of 0.5 and 1 respectively.